МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)

Факультет компьютерных наук и прикладной математики

Кафедра вычислительной математики и программирования

**КУРСОВОЙ ПРОЕКТ**

**по курсу**

**«Вычислительные системы»**

**Задание 4**

«Процедуры и функции в качестве параметров»

**Выполнил:**

студент группы М8О-108Б-22

Формалёв Александр Сергеевич

**Преподаватель:**

Сахарин Никита Александрович

Дата:

Подпись:

Москва, 2022

**Цель работы**

Составить программу на языке Си с процедурами решения трансцендентных алгебраических уравнений различными численными методам (итераций, Ньютона и половинного деление — дихотомии). Уравнения оформить как функции параметры, разрешив относительно неизвестной величины в случае необходимости. Применить каждую процедуру к решению уравнения — заданного вариантом. Если метод неприменим, дать математическое обоснование и графическую иллюстрацию, например, с использованием gnuplot.

**Общий метод решения**

Необходимо описать возможностями языка Си вычисление корня функции методом итерации, дихотомии (половинного деления), Ньютона. Далее нужно сравнить полученное значение со значением в таблице.

**Задание**

**Уравнение:**

**Отрезок, содержащий корни:**

**Приближённое значение корня:**

**Базовый метод:** метод Ньютона

**Теоретическая часть**

**Метод Ньютона:** Метод Ньютона является частным случаем метода итераций. Условие сходимости метода: на отрезке [a, b]. Итерационный процесс: .

**Метод итераций:** Идея метода заключается в замене исходного уравнения уравнением вида . Достаточное условие сходимости метода: . Это условие необходимо проверить перед началом решения задачи, так как функция может быть выбрана неоднозначно, причём в случае неверного выбора указанной функции метод расходится. Начальное приближение корня: . Итерационный процесс: . Условие окончания: . Приближённое значение корня: .

**Код программы**

#include <stdio.h>

#include <math.h>

#include <assert.h>

#define EPS (0.0001)

typedef double dbl;

dbl myabs(dbl x);

dbl F(dbl x);

dbl DF(dbl x);

dbl DDF(dbl x);

void Newton(dbl x0);

int main(){

int a = 1, b = 2;

assert(myabs(F(b) \* DDF(b)) < DF(b) \* DF(b));

Newton((b - a) / 2.0);

return 0;

}

dbl F(dbl x){

return 0.1 \* x \* x - x \* log(x);

}

dbl myabs(dbl x){

if(x < 0){

return -1 \* x;

} else{

return x;

}

}

dbl DF(dbl x){

return 0.2 \* x - log(x) - 1;

}

dbl DDF(dbl x){

return 0.2 - (1 / x);

}

void Newton(dbl x0){

dbl x1 = x0, x2 = x1 - (F(x1) / DF(x1));

while(myabs(x1 - x2) >= EPS){

x1 = x2;

x2 = x1 - (F(x1) / DF(x1));

}

printf("%.4lf\n", x2);

}

**Вывод**

Работа над данным курсовым проектом помогла познакомиться с базовыми численными методами: итераций, Ньютона и дихотомии. Были более подробно изучены нецелочисленные типы данных в языке Си, такие как double, long double и float.